# 18.IP分片及重组库

IP分段和重组库实现IPv4和IPv6报文的分片和重组。

## 18.1.报文分片

报文分段例程将输入报文划分成多个分片。rte\_ipv4\_fragment\_packet()和rte\_ipv6\_fragment\_packet()函数都假定输入mbuf数据指向报文的IP报头的开始（即L2报头已经被剥离）。为了避免复制实际数据包的数据，使用零拷贝技术（rte\_pktmbuf\_attach）。对于每个片段，将创建两个新的mbuf：

* Direct mbuf：mbuf将包含新片段的L3头部。
* Indirect mbuf：源数据包附加到mbuf。数据字段指向原始数据包数据的附加数据偏移量开始处。

然后将L3头部从原始mbuf复制到“direct”mbuf并更新以反映新的碎片状态。 请注意，对于IPv4，不会重新计算头校验和，其值设置为零。

最后，通过mbuf的下next字段将每个片段的“dirext”和“indirect”mbuf链接在一起，以构成新片段的数据包。

调用者可以明确指定哪些mempools应用于从中分配“direct”和“indirect”mbufs。

有关direct和indirect mbufs的信息，请参阅直接和间接缓冲区。

## 18.2.报文重组

### 18.2.1.IP分片表

报文分片表中维护已经接收到的数据包片段的信息。

每个IP数据包由三个字段：<源IP地址>，<目标IP地址>，<ID>唯一标识。

请注意，报文分片表上的所有更新/查找操作都不是线程安全的。因此，如果不同的执行上下文（线程/进程）要同时访问同一个表，那么必须提供一些外部同步机制。

每个表项可以保存最多RTE\_LIBRTE\_IP\_FRAG\_MAX（默认值为4）片段的数据包的信息。

代码示例，演示了创建新的片段表：

frag\_cycles = (rte\_get\_tsc\_hz() + MS\_PER\_S - 1) / MS\_PER\_S \* max\_flow\_ttl;

bucket\_num = max\_flow\_num + max\_flow\_num / 4;

frag\_tbl = rte\_ip\_frag\_table\_create(max\_flow\_num, bucket\_entries, max\_flow\_num, frag\_cycles, socket\_id);

内部片段表是一个简单的哈希表。 基本思想是使用两个哈希函数和<bucket\_entries> \*关联性。 这为每个Key在散列表中提供了2 \* <bucket\_entries>可能的位置。当发生冲突并且所有2 \* <bucket\_entries>都被占用时，ip\_frag\_tbl\_add()只是返回失败，而不是将现有的Key重新插入到另外的位置。

此外，驻留在表中的条目如果比<max\_cycles>更长，被认为是无效的，可以被新的条目删除/替换。

请注意，重新组合需要分配很多mbuf。在任何给定时间（2 \* bucket\_entries \* RTE\_LIBRTE\_IP\_FRAG\_MAX \* <每个数据包的最大mbufs数>>）可以存储在等待剩余片段的Fragment Table中。

### 18.2.2.报文重组

报文分组处理和重组由rte\_ipv4\_frag\_reassemble\_packet()/rte\_ipv6\_frag\_reassemble\_packet()完成。它们返回一个指向有效mbuf的指针，它包含重新组合的数据包，或者返回NULL（如果数据包由于某种原因而无法重新组合）。

这些功能包括：

1. 搜索片段表，输入数据包的<IPv4源地址，IPv4目的地址，数据包ID>。
2. 如果找到该条目，则检查该条目是否已经超时。如果是，则释放所有以前收到的碎片，并从条目中删除有关它们的信息。
3. 如果没有找到这样的Key的条目，那么尝试通过以下两种方法之一创建一个新的：  
   a) 用作空条目。  
   b) 删除一个超时条目，与它mbufs关联的空闲mbufs，并在其中存储一个带有指定键的新条目。
4. 使用新的片段信息更新条目，并检查是否可以重新组合数据包（数据包的条目包含所有片段）。  
   a) 如果是，则重新组装数据包，将表的条目标记为空，并将重新组装的mbuf返回给调用者。  
   b) 如果否，则向调用者返回一个NULL。

如果在分组处理的任何阶段遇到错误（例如：不能将新条目插入片段表或无效/超时片段），则该函数将释放所有与分组片段相关联的标记表条目 作为无效并将NULL返回给调用者。

18.2.3. 调试日志及统计收集  
RTE\_LIBRTE\_IP\_FRAG\_TBL\_STAT配置宏用于控制片段表的统计信息收集。默认情况下未启用。

RTE\_LIBRTE\_IP\_FRAG\_DEBUG控制IP片段处理和重组的调试日志记录。默认情况下禁用。请注意，在日志记录包含大量详细信息时，会减慢数据包处理速度，并可能导致丢失大量数据包。